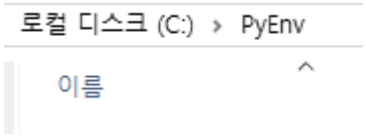
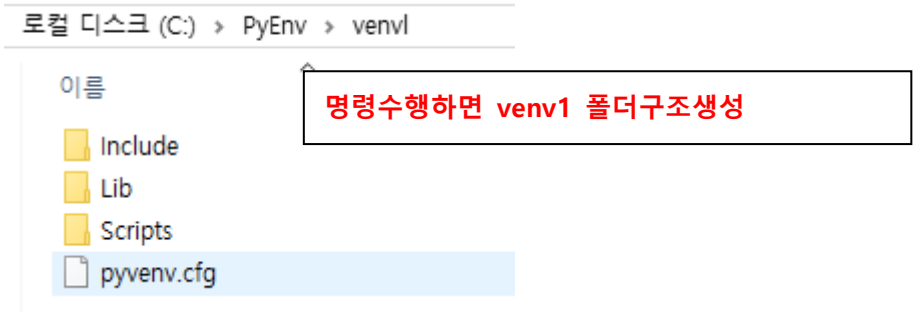
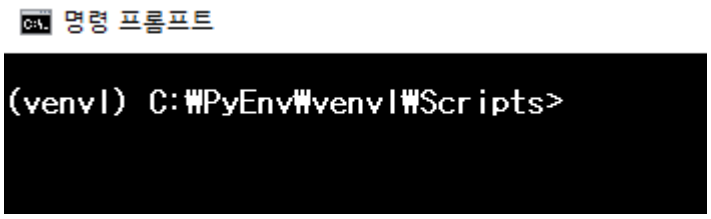
1.가상환경 만들 폴더 생성



2. C:₩Users₩USER₩AppData₩Local₩Programs₩Python₩Python36-32>
python Tools/Scripts/pyvenv.py c:/PyEnv/venvl



3. C:₩PyEnv₩venvl₩Scripts>activate
윈도우즈에서 가상환경을 활성화(activate)



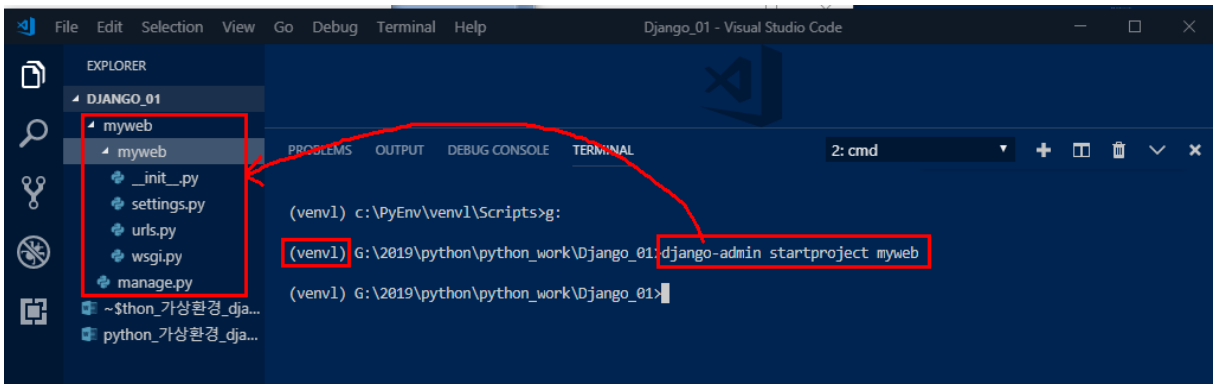
종료는 exit

4. (venvl) C:₩PyEnv₩venvl₩Scripts>python -m pip install --upgrade pip
pip upgrade

5. (venvl) C:₩PyEnv₩venvl₩Scripts>pip install django==2.1
django 2.1 버전 설치

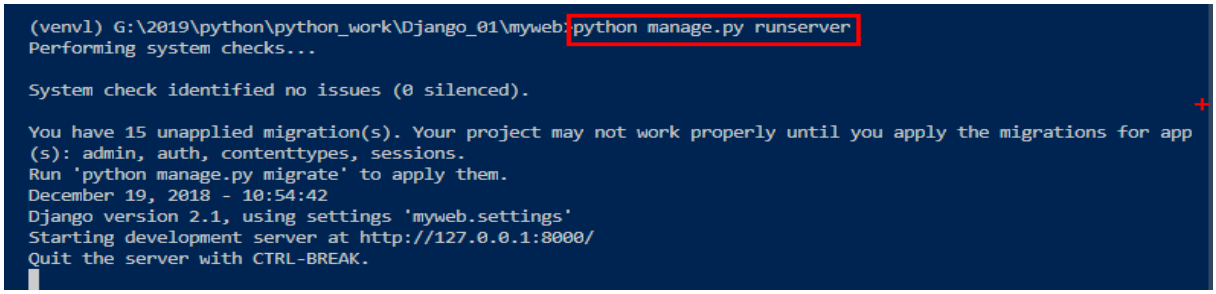
6. (venvl) G:₩2019₩python₩python_work₩Django_01>**django-admin startproject myweb**

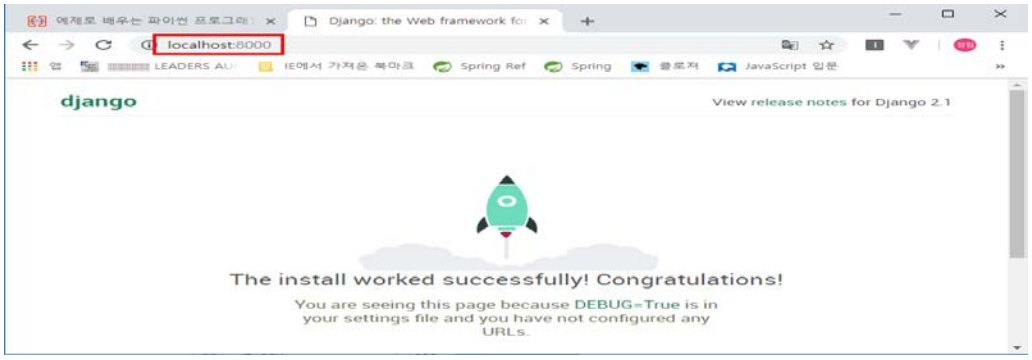
프로젝트를 만들 디렉토리로 이동한 후, 아래와 같이 "django-admin startproject 프로젝트명" 를 실행하여 새 프로젝트를 생성한다.



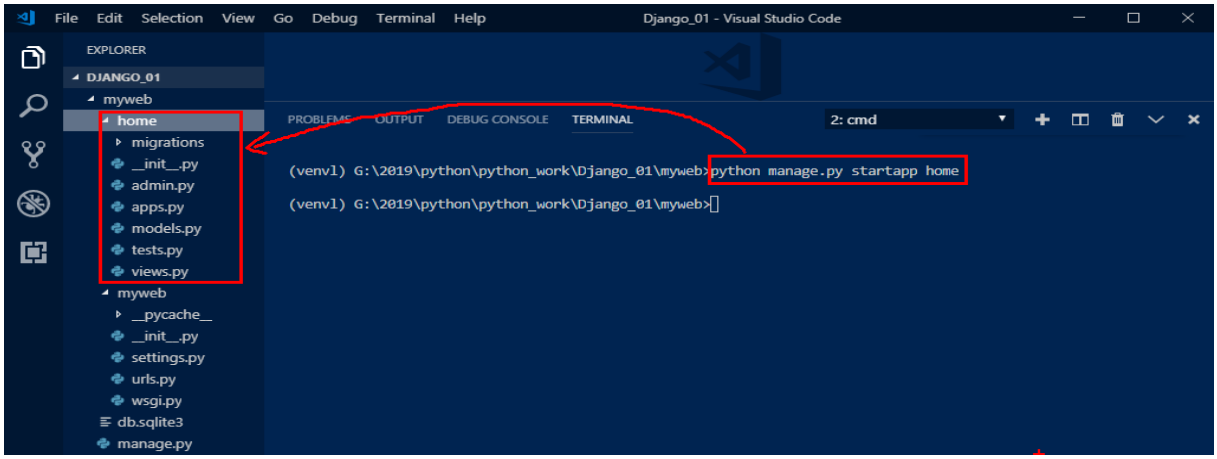
C:₩PyEnv₩venvl₩Scripts>**activate** 윈도우즈에서 가상환경을 활성화(activate)

7. (venvl) G:₩2019₩python₩python_work₩Django_01₩myweb>**python manage.py runserver**

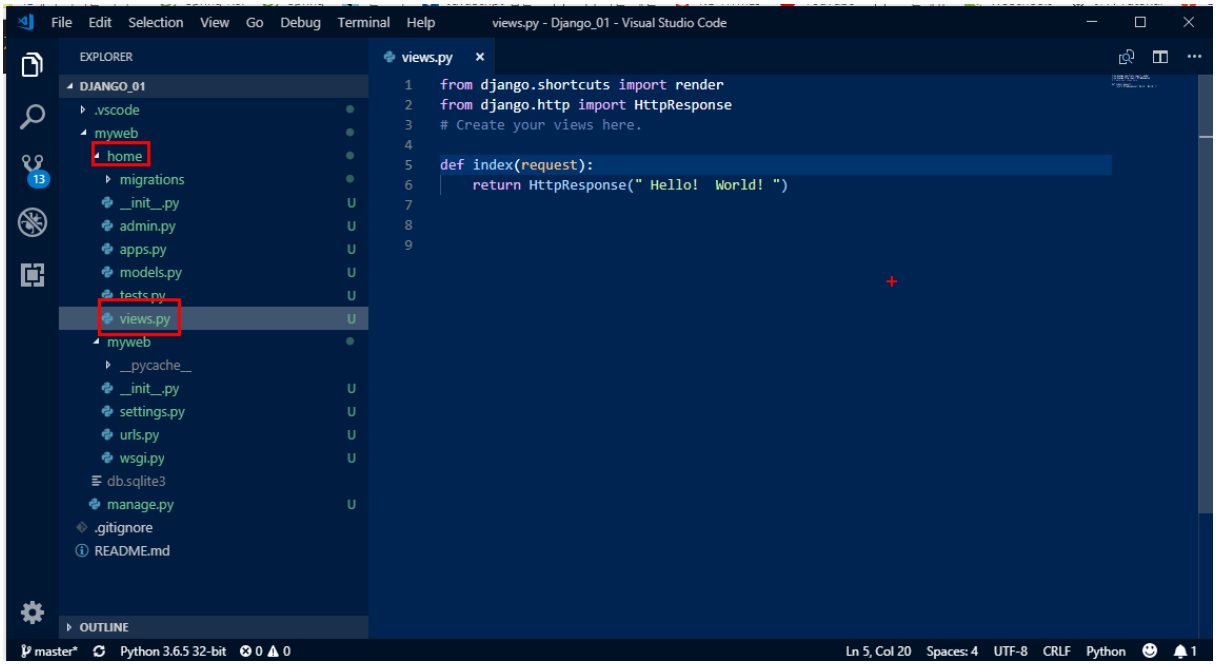


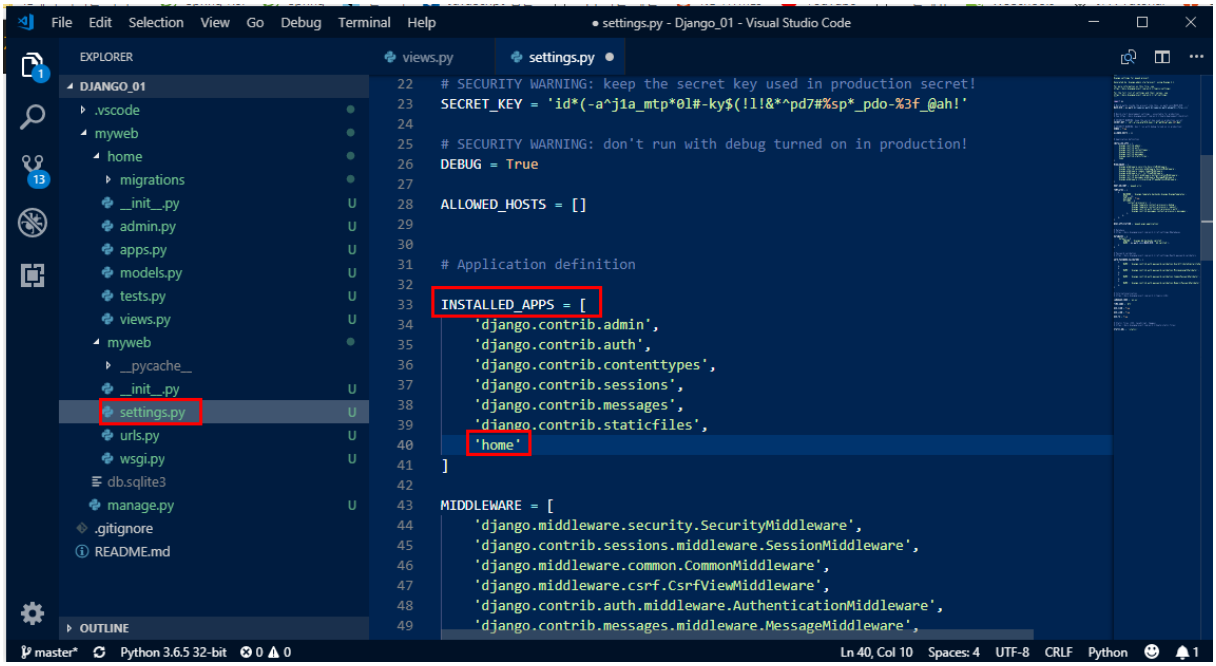


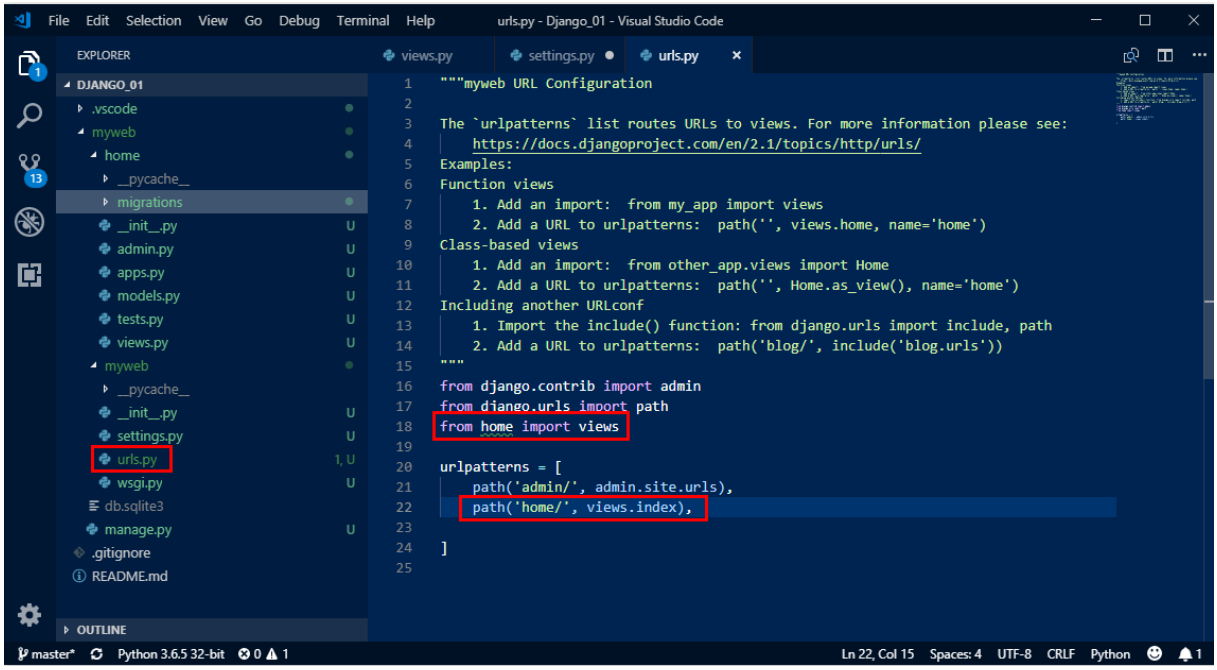
8. (venvl) G:₩2019~~_work₩Django_01₩myweb>**python manage.py startapp home**



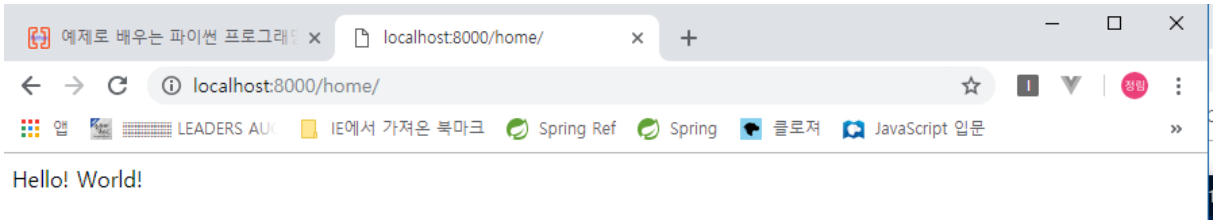
https://github.com/javains/Django_01.git 연동



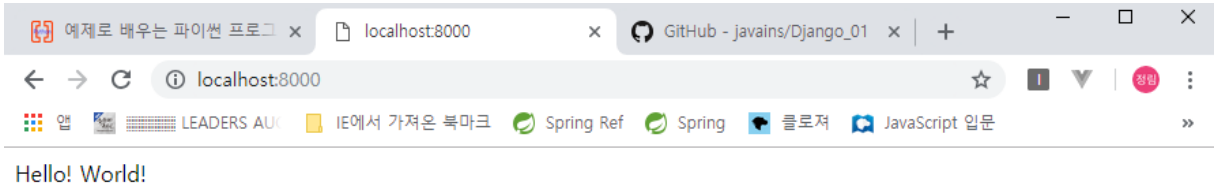




(venvl) G:₩2019₩python₩django_work₩Django_01₩myweb>python manage.py runserver

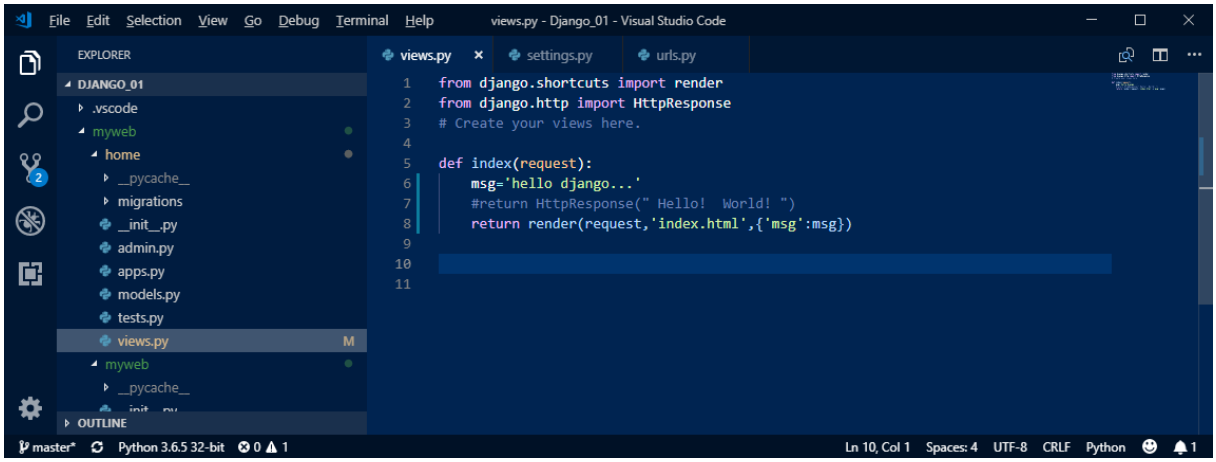


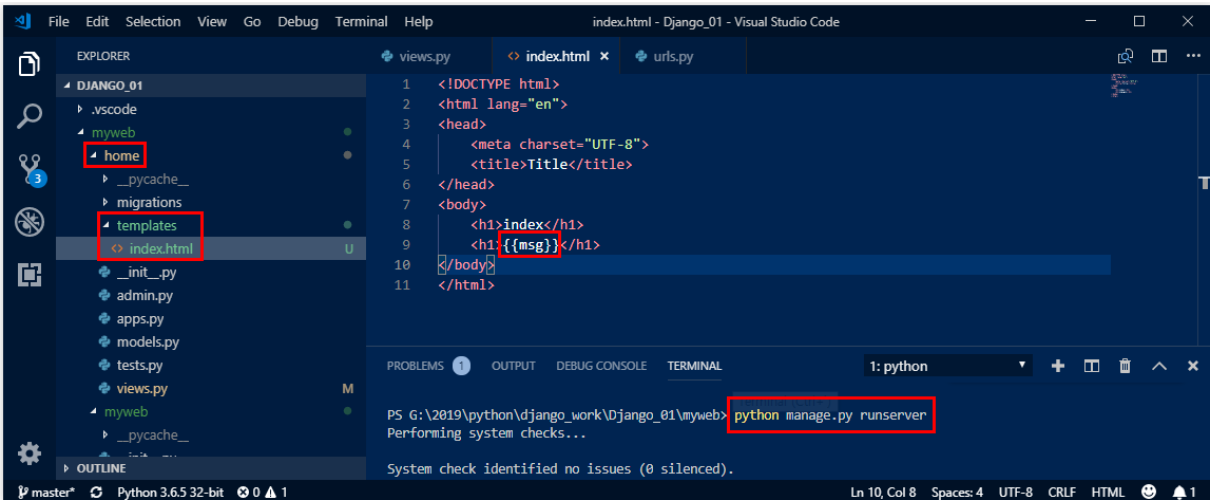

```
urlpatterns = [
    path('admin/', admin.site.urls),
    path('', views.index),
]
```

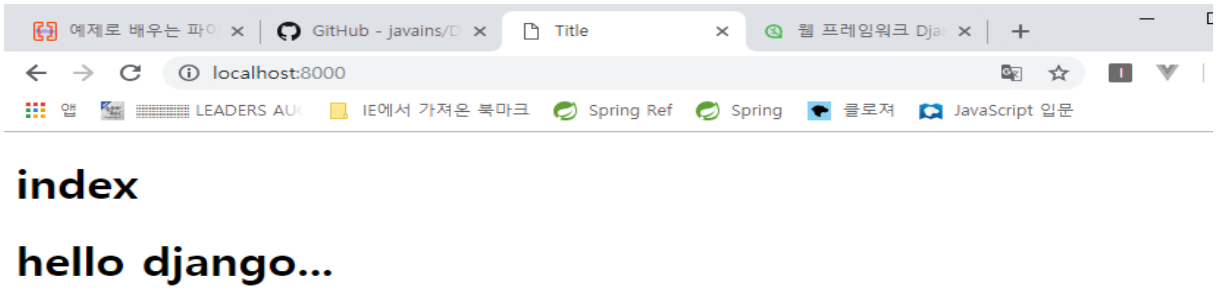


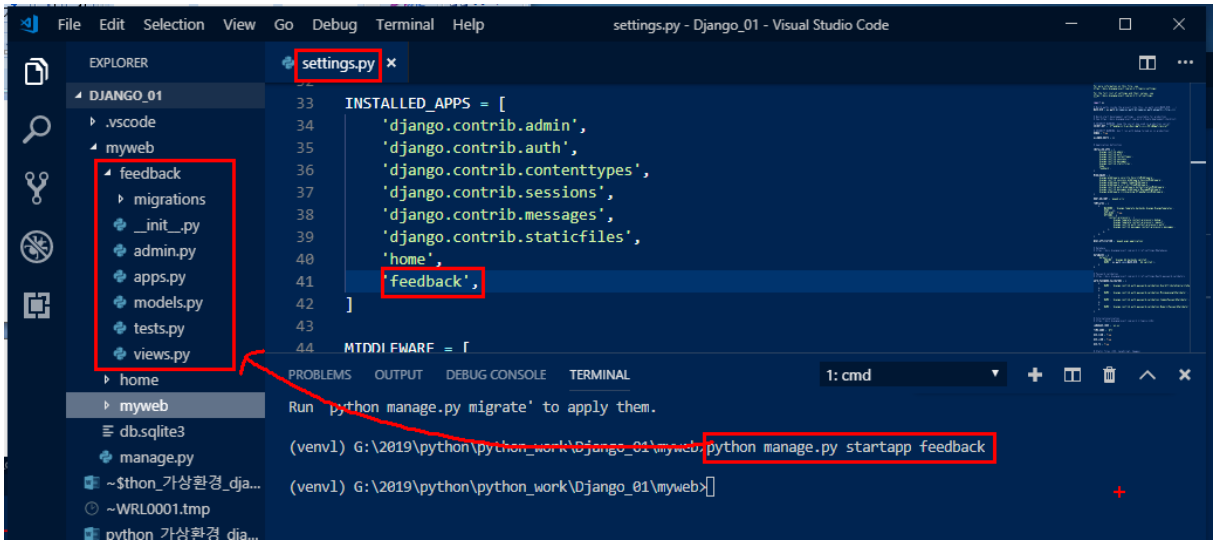
템플릿 (Template)은 View로부터 전달된 데이타를 템플릿에 적용하여 Dynamic 한 웹페이지를 만드는데 사용된다.

Django 개발 가이드라인은 "App폴더/templates/템플릿파일" 처럼, 각 App 폴더 밑에 templates 서브폴더를 만들고 다시 그 안에 App명을 사용하여 서브폴더를 만든 후 템플릿 파일을 그 안에 넣기를 권장한다 (예: /home/templates/index.html).

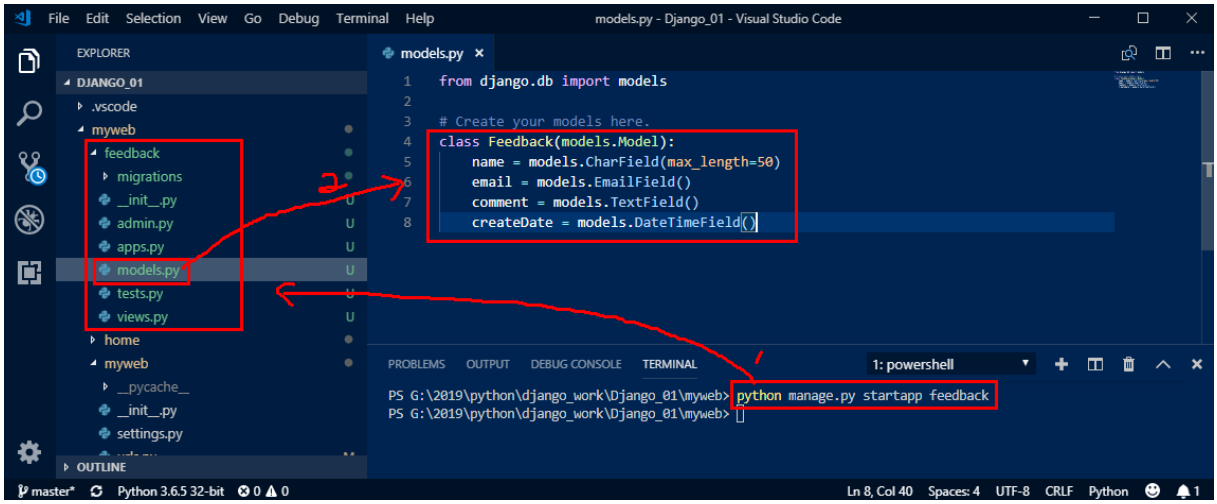

File Edit Selection View Go Debug Terminal Help
views.py - Django_01 - Visual Studio Code
EXPLORER
DJANGO_01
.vscode
myweb
home
__pycache__
migrations
__init__.py
admin.py
apps.py
models.py
tests.py
views.py
myweb
__pycache__
OUTLINE
views.py
settings.py
urls.py
from django.shortcuts import render
from django.http import HttpResponse
# Create your views here.
def index(request):
msg='hello django...'
#return HttpResponse(" Hello! World! ")
return render(request,'index.html',{'msg':msg})
master* Python 3.6.5 32-bit 0 1
Ln 10, Col 1 Spaces: 4 UTF-8 CRLF Python 1


File Edit Selection View Go Debug Terminal Help
index.html - Django_01 - Visual Studio Code
EXPLORER
DJANGO_01
.vscode
myweb
home
__pycache__
migrations
templates
index.html
__init__.py
admin.py
apps.py
models.py
tests.py
views.py
myweb
__pycache__
OUTLINE
views.py
index.html
urls.py
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<title>Title</title>
</head>
<body>
<h1>index</h1>
<h1>{{msg}}</h1>
</body>
</html>
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
1: python
PS G:\2019\python\django_work\Django_01\myweb> python manage.py runserver
Performing system checks...
System check identified no issues (0 silenced).
master* Python 3.6.5 32-bit 0 1
Ln 10, Col 8 Spaces: 4 UTF-8 CRLF HTML 1


예제로 배우는 파
GitHub - javains/D
Title
웹 프레임워크 Dja
localhost:8000
앱
LEADERS AU
IE에서 가져온 북마크
Spring Ref
Spring
클로저
JavaScript 입문
index
hello django...



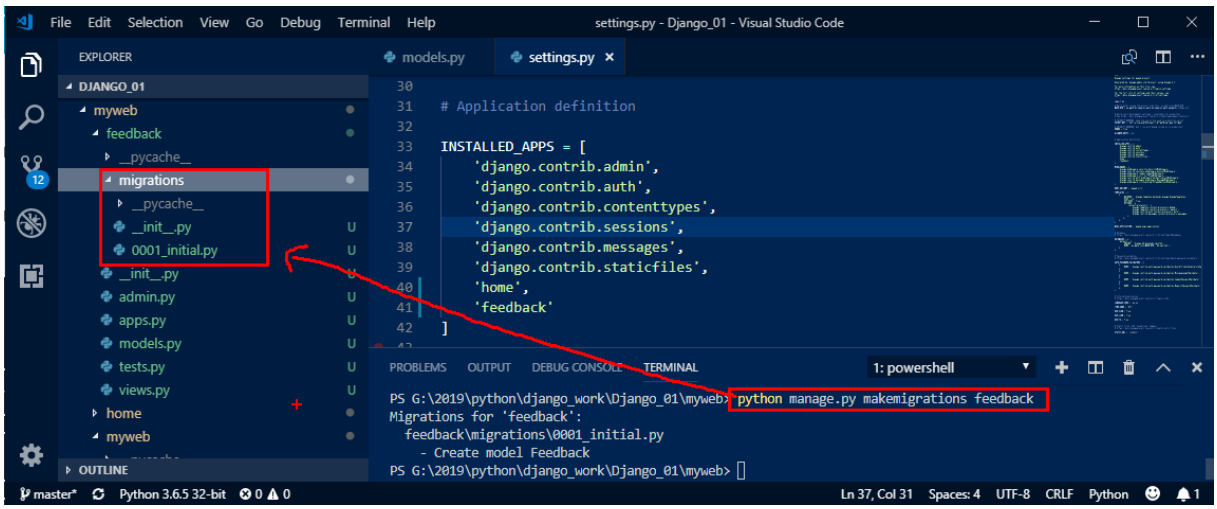
Django 모델 (Model)




```python
class Feedback(models.Model):
    name = models.CharField(max_length=50)
    email = models.EmailField()
    comment = models.TextField(null=True)
    createDate = models.DateTimeField(auto_now_add=True)
```

settings.py => INSTALLED_APPS 리스트에 'feedback'

DB Migration

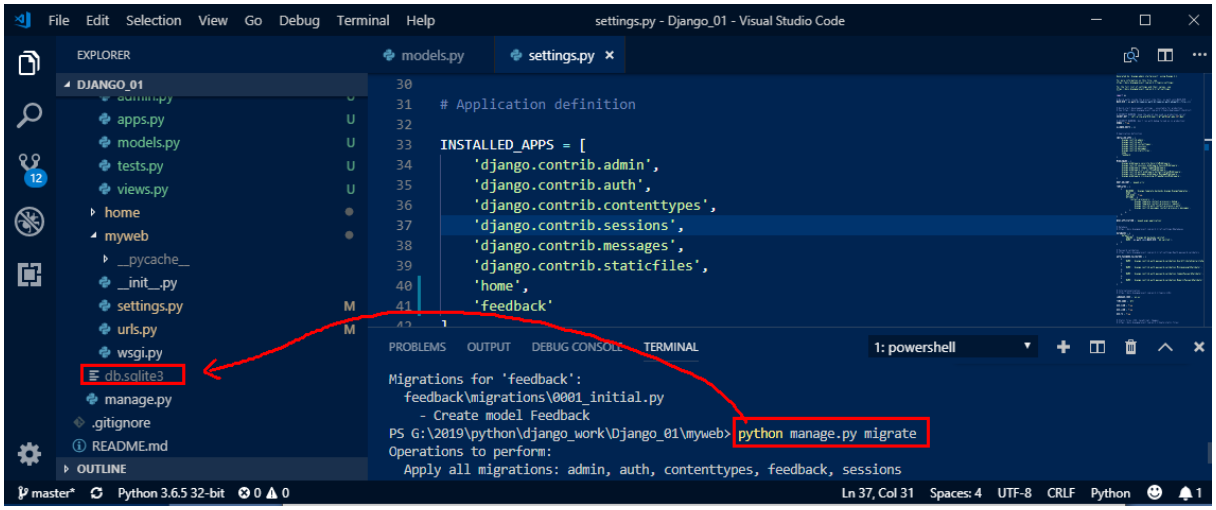


1. PS G:₩~~~_work₩Django_01₩myweb> python manage.py makemigrations feedback

==> feedback app 에 model 변화가 있는지 확인만한다.

2. PS G:₩2019₩python₩django_work₩Django_01₩myweb> python manage.py migrate

==> model을 DataBase에 적용한다.

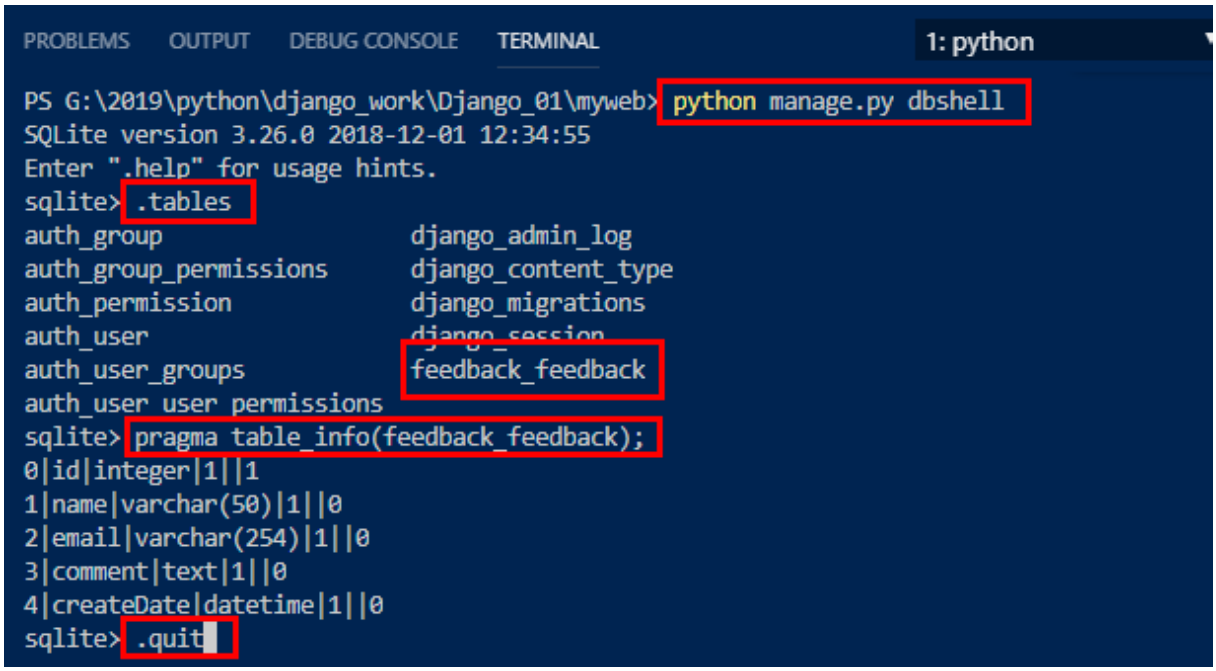


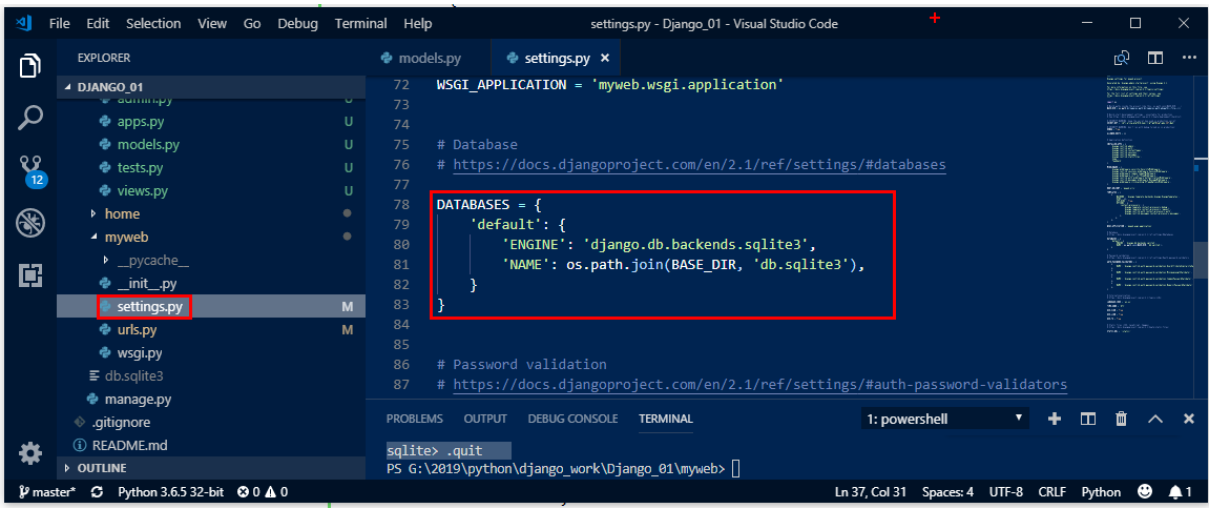
G:₩2019₩python₩django_work₩Django_01₩myweb> python manage.py dbshell

sqlite> .tables

sqlite> pragma table_info(feedback_feedback);

sqlite> .quit

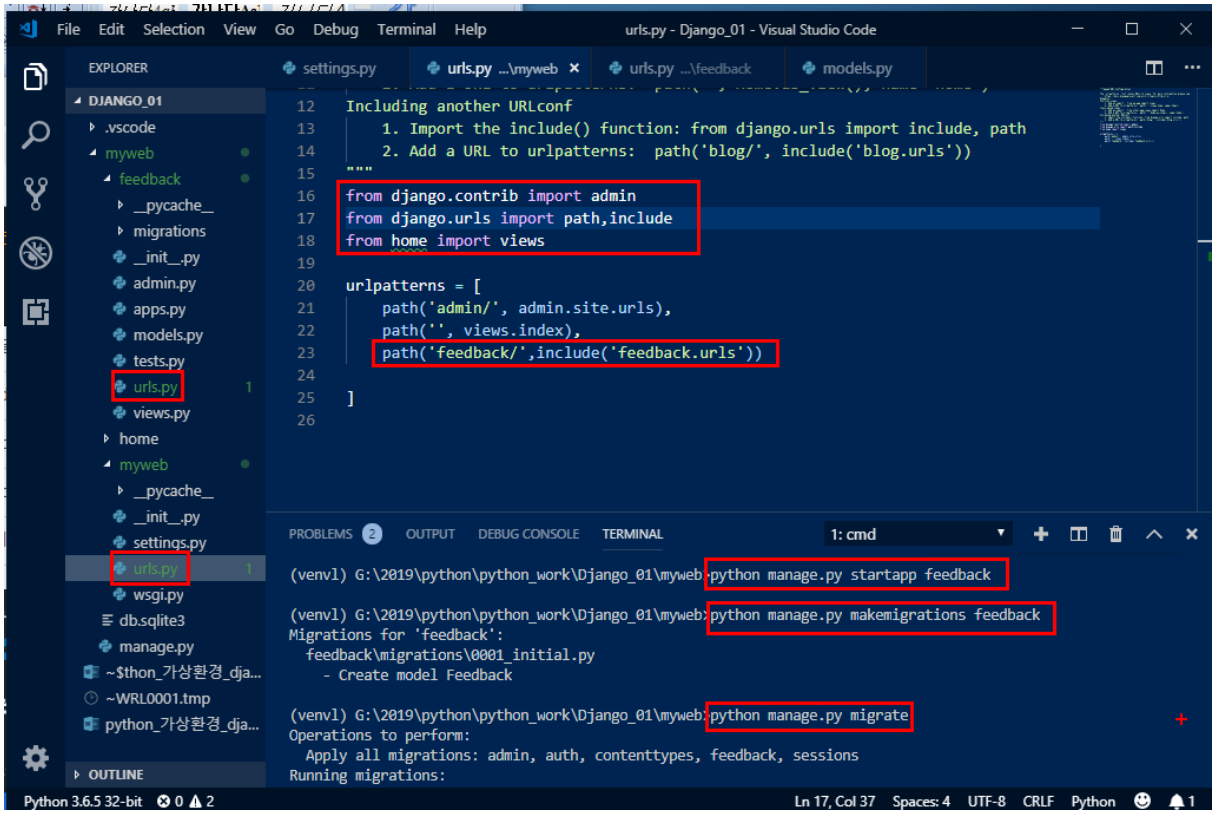





```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'MyDB',
        'USER': 'user1',
        'PASSWORD': 'pwd',
        'HOST': 'localhost',
        'PORT': '3306',
    }
}
```

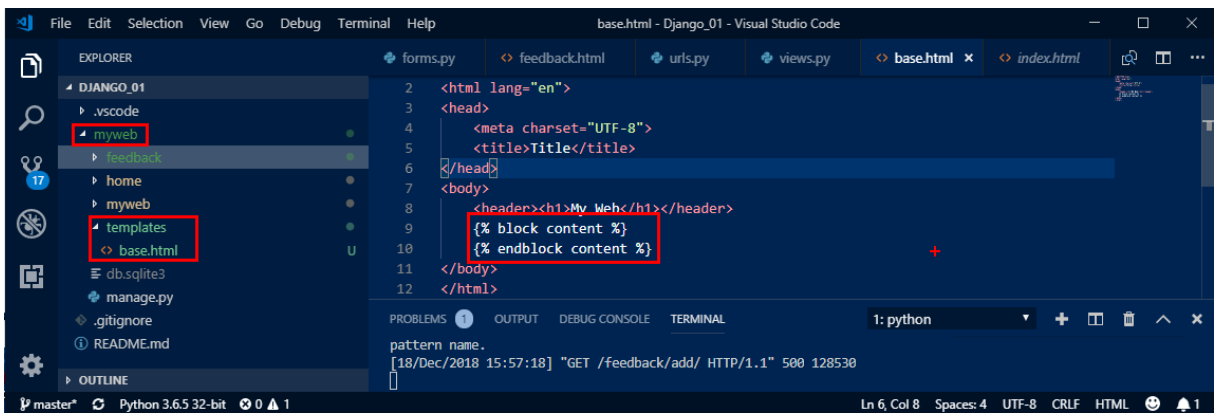
- django.db.backends.postgresql
- django.db.backends.mysql
- django.db.backends.sqlite3
- django.db.backends.oracle

```
from feedback.models import *
from datetime import datetime

        INSERT
# Feedback 객체 생성
fb = Feedback(name = 'Kim', email = 'kim@test.com', comment='Hi',
createDate=datetime.now())
fb.save()      # 새 객체 INSERT

        SELECT
for f in Feedback.objects.all():
    s += str(f.id) + ' : ' + f.name + '₩n'

row = Feedback.objects.get(pk=1)
print(row.name)

rows = Feedback.objects.filter(name='Kim')
n = Feedback.objects.count()    # 데이타의 갯수(row 수)
rows = Feedback.objects.distinct('name')
rows = Feedback.objects.order_by('name').first()
rows = Feedback.objects.order_by('name').last()


        UPDATE
fb = Feedback.objects.get(pk=1)
fb.name = 'Park'
fb.save()

        DELETE
fb = Feedback.objects.get(pk=2)
fb.delete()
```



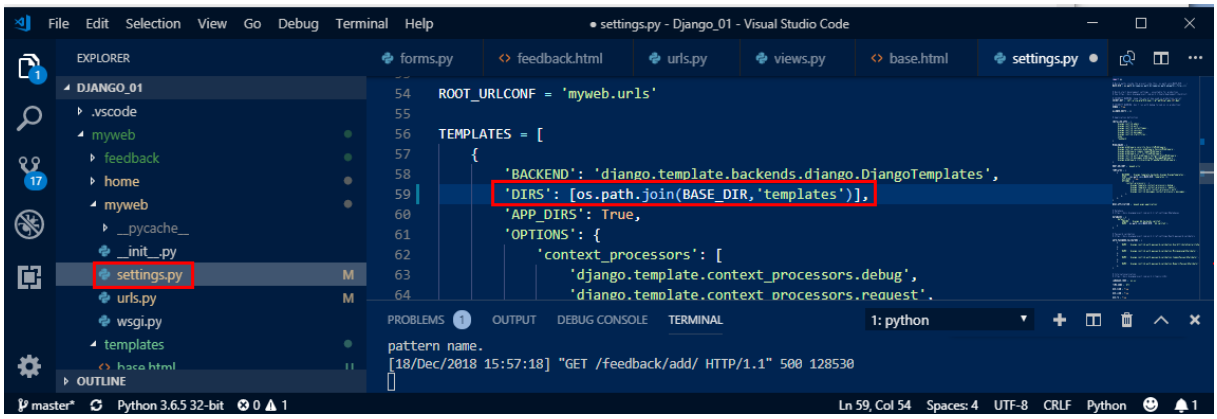

```
from django.contrib import admin
from django.urls import path
from feedback import views
urlpatterns = [
    path('add/', views.add),
]
```

```
urlpatterns = [
    path('add/', views.add),
    path('update/', views.update),
    path('list/', views.list),
]
```

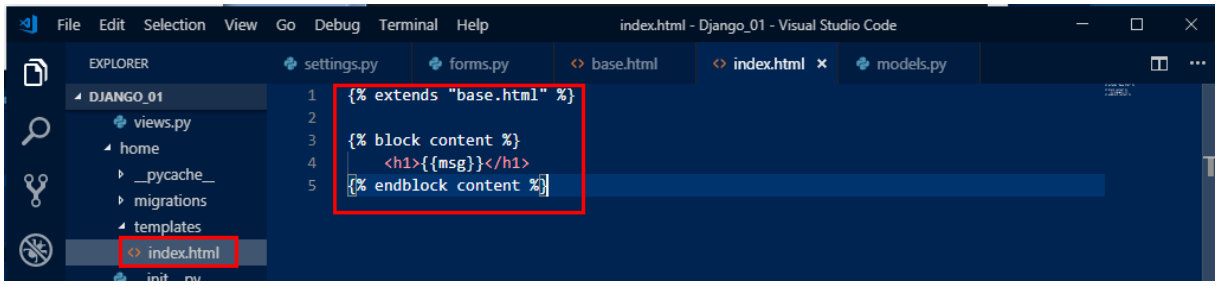


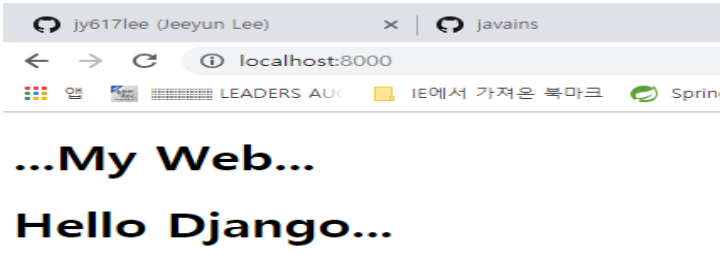
./templates/base.html 이라는 Base 템플릿을 만들었는데,
이 파일 안에 각 웹페이지에서 변경 혹은 삽입할 영역을 {% block 블럭명 %} 으로 지정한다. 여기서는 블럭명을 content로 정하여 {% block content %} 으로 표시

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
</head>
<body>
    <header>...My Web...</header>
    {% block content %}
    {% endblock content %}
</body>
</html>
```

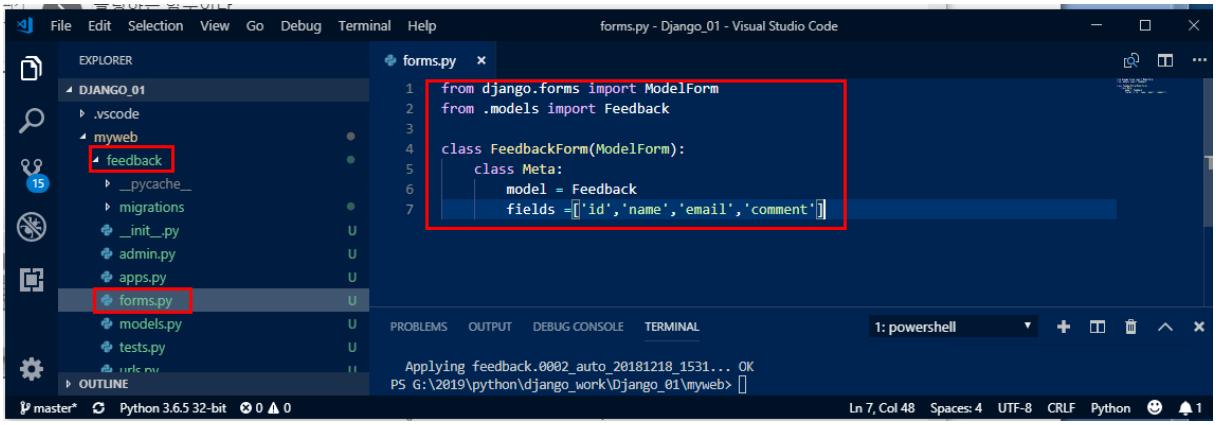


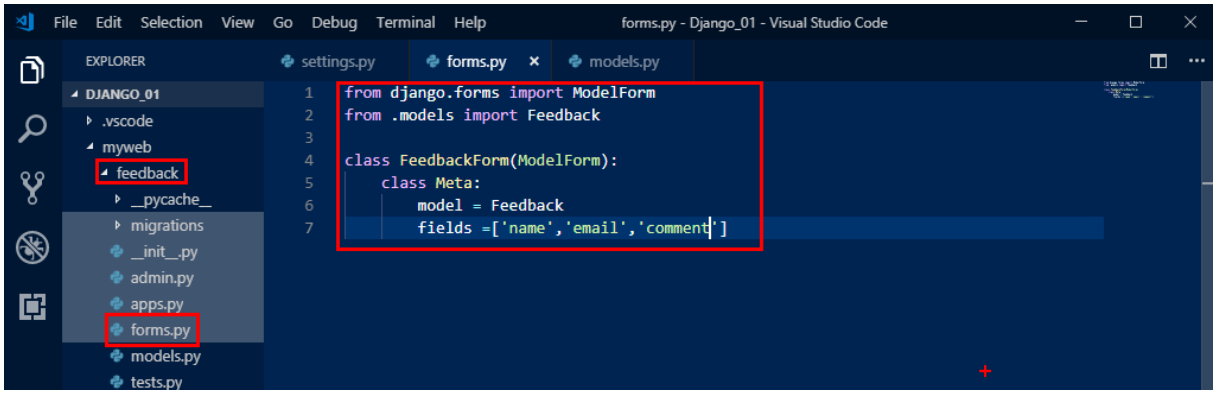
템플릿 위치 셋팅



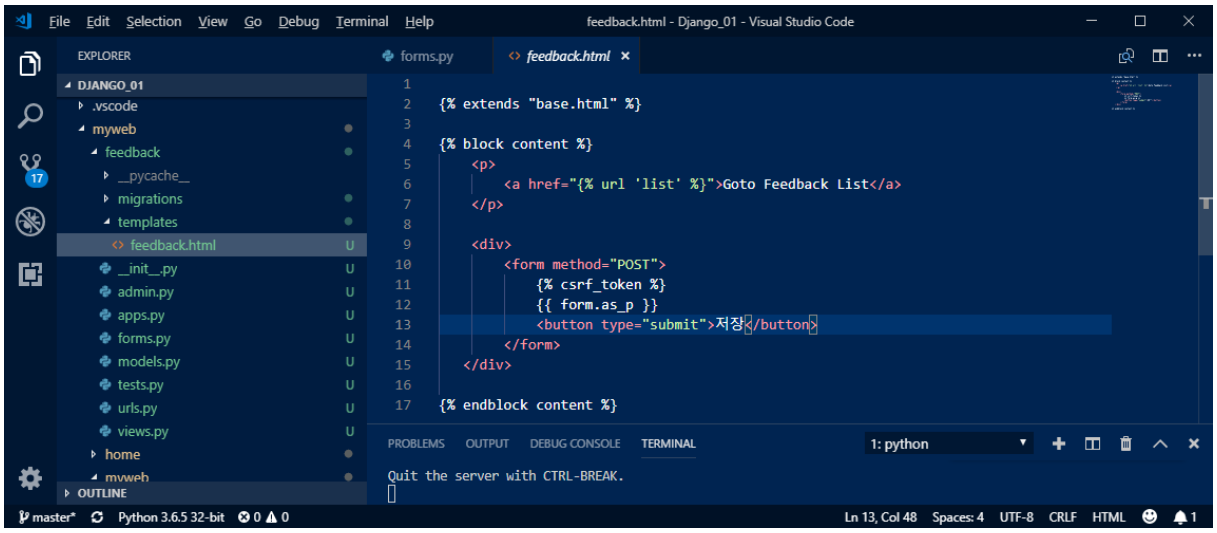


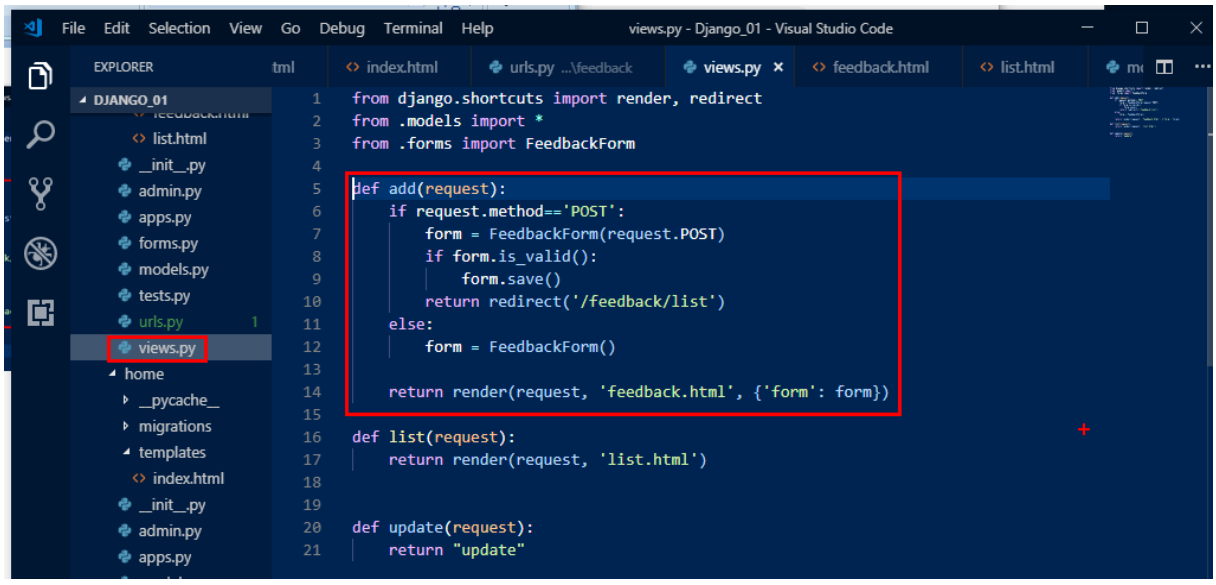
Django 폼 (Form)

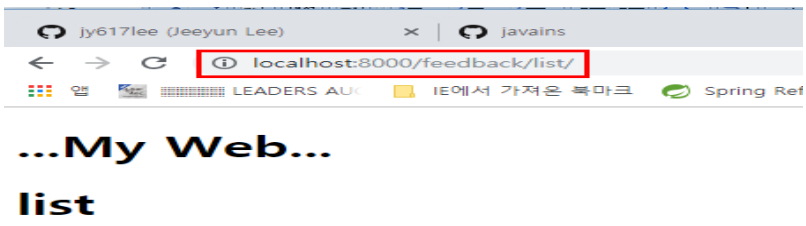


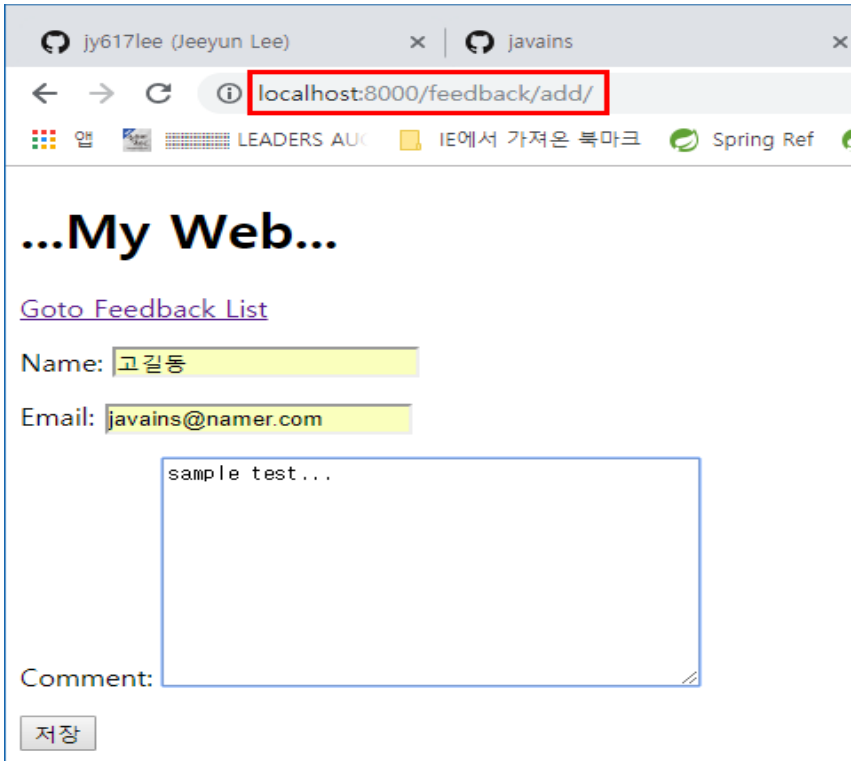


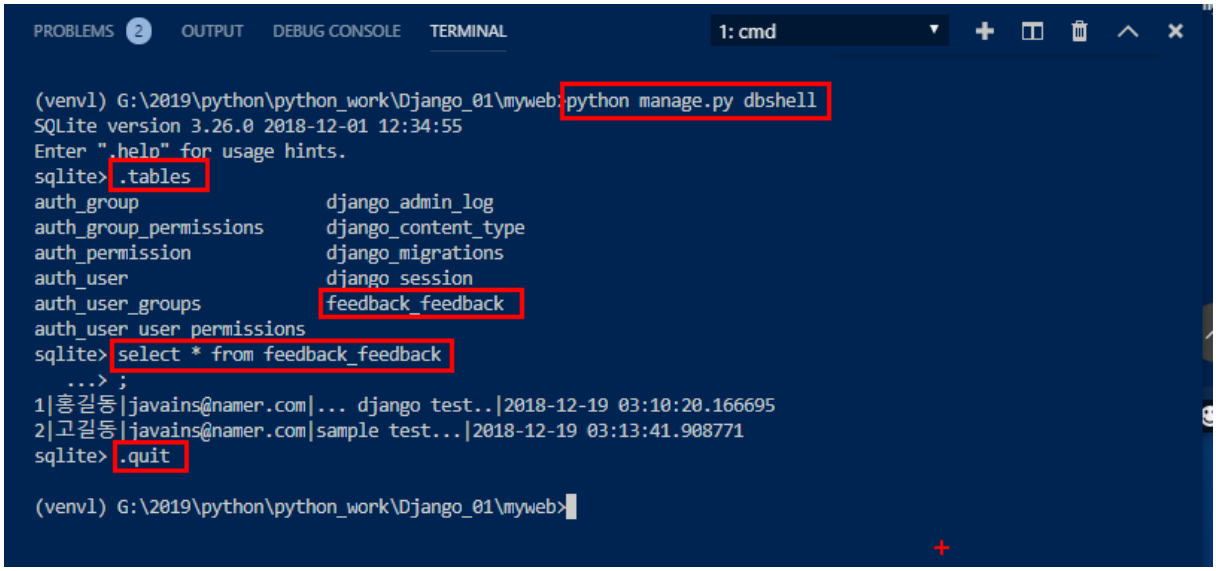
CSRF (Cross Site Request Forgeries)는 웹 해킹 기법의 하나로 **Django**는 이를 방지하기 위한 기능을 기본적으로 제공하고 있다. **Django**에서 **HTTP POST, PUT, DELETE**을 할 경우 이 태그를 넣어 주어야 한다

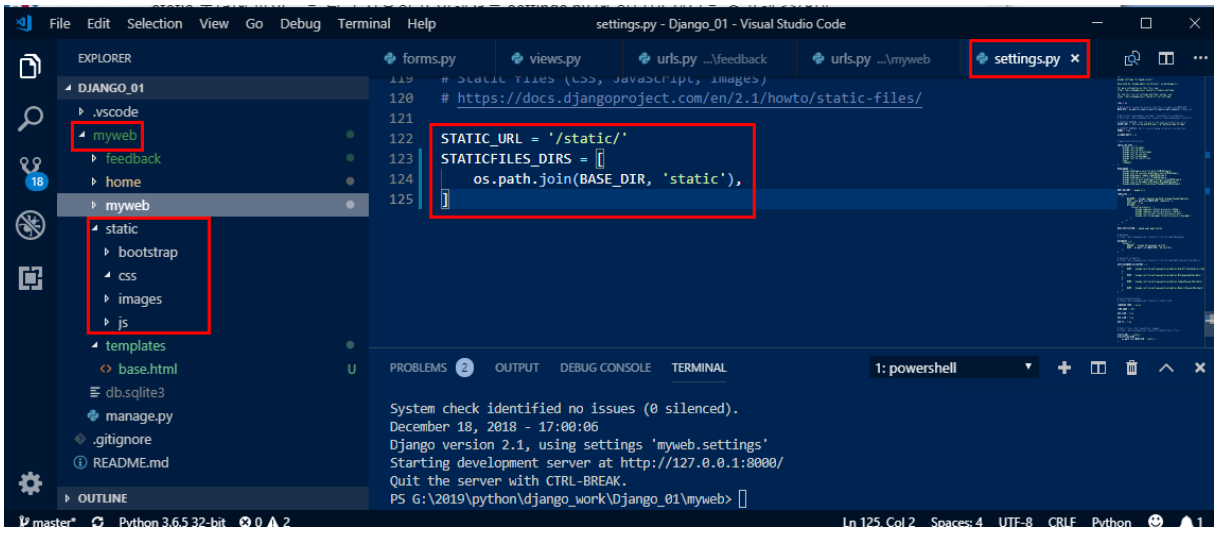


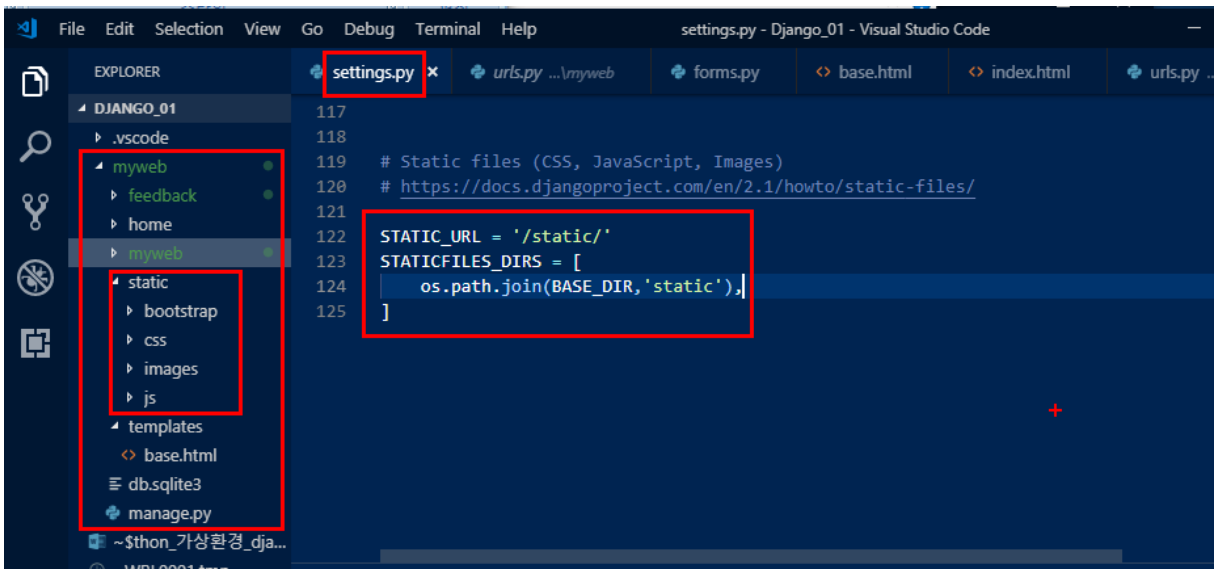

File Edit Selection View Go Debug Terminal Help
views.py - Django_01 - Visual Studio Code
EXPLORER
DJANGO_01
list.html
__init__.py
admin.py
apps.py
forms.py
models.py
tests.py
urls.py
views.py
home
__pycache__
migrations
templates
index.html
__init__.py
admin.py
apps.py
index.html
urls.py ...\feedback
views.py
feedback.html
list.html
from django.shortcuts import render, redirect
from .models import *
from .forms import FeedbackForm
def add(request):
if request.method=='POST':
form = FeedbackForm(request.POST)
if form.is_valid():
form.save()
return redirect('/feedback/list')
else:
form = FeedbackForm()
return render(request, 'feedback.html', {'form': form})
def list(request):
return render(request, 'list.html')
def update(request):
return "update"


jy617lee (Jeeyun Lee)
javains
localhost:8000/feedback/list/
앱
LEADERS AU
IE에서 가져온 북마크
Spring Ref
...My Web...
list


jy617lee (Jeeyun Lee)
javains
localhost:8000/feedback/add/
앱
LEADERS AU
IE에서 가져온 북마크
Spring Ref
...My Web...
Goto Feedback List
Name: 고길동
Email: javains@namer.com
sample test...
Comment:
저장


PROBLEMS 2
OUTPUT
DEBUG CONSOLE
TERMINAL
1: cmd
(venvl) G:\2019\python\python_work\Django_01\myweb>python manage.py dbshell
SQLite version 3.26.0 2018-12-01 12:34:55
Enter ".help" for usage hints.
sqlite> .tables
auth_group
auth_group_permissions
auth_permission
auth_user
auth_user_groups
auth_user_user_permissions
django_admin_log
django_content_type
django_migrations
django_session
feedback_feedback
sqlite> select * from feedback_feedback
...> ;
1|홍길동|javains@namer.com|... django test..|2018-12-19 03:10:20.166695
2|고길동|javains@namer.com|sample test...|2018-12-19 03:13:41.908771
sqlite> .quit
(venvl) G:\2019\python\python_work\Django_01\myweb>

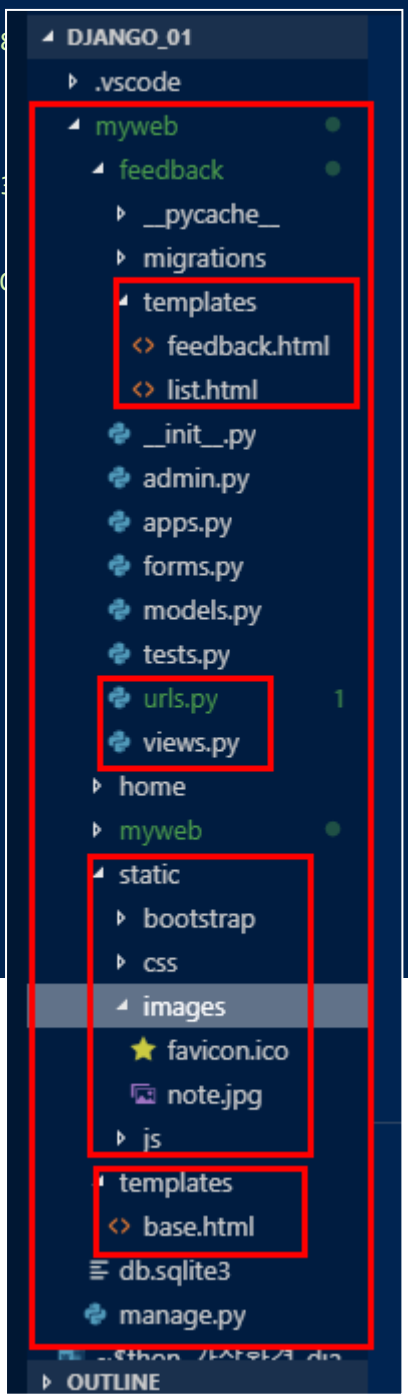

File Edit Selection View Go Debug Terminal Help
settings.py - Django_01 - Visual Studio Code
EXPLORER
DJANGO_01
.vscode
myweb
feedback
home
myweb
static
bootstrap
css
images
js
templates
base.html
db.sqlite3
manage.py
.gitignore
README.md
OUTLINE
forms.py
views.py
urls.py ...\feedback
urls.py ...\myweb
settings.py
# https://docs.djangoproject.com/en/2.1/howto/static-files/
STATIC_URL = '/static/'
STATICFILES_DIRS = [
os.path.join(BASE_DIR, 'static'),
]
PROBLEMS 2
OUTPUT
DEBUG CONSOLE
TERMINAL
1: powershell
System check identified no issues (0 silenced).
December 18, 2018 - 17:00:06
Django version 2.1, using settings 'myweb.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
PS G:\2019\python\django_work\Django_01\myweb>


File Edit Selection View Go Debug Terminal Help
settings.py - Django_01 - Visual Studio Code
EXPLORER
DJANGO_01
.vscode
myweb
feedback
home
myweb
static
bootstrap
css
images
js
templates
base.html
db.sqlite3
manage.py
~$thon_가상환경_dja...
settings.py
urls.py ...\myweb
forms.py
base.html
index.html
urls.py
# Static files (CSS, JavaScript, Images)
# https://docs.djangoproject.com/en/2.1/howto/static-files/
STATIC_URL = '/static/'
STATICFILES_DIRS = [
os.path.join(BASE_DIR,'static'),
]

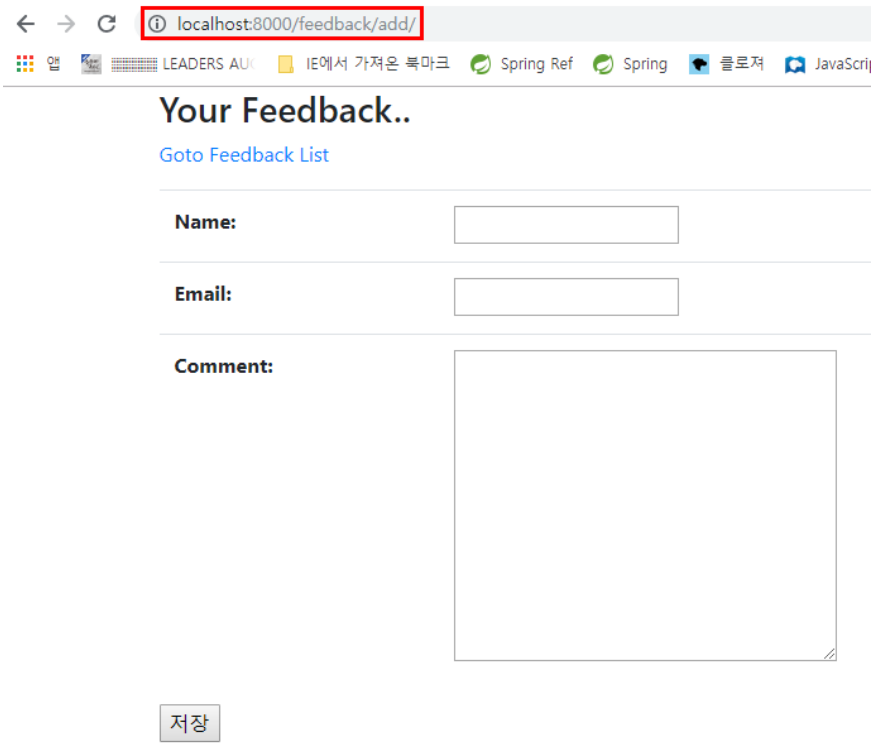
```
from django.contrib import admin
from django.urls import path
from feedback import views

urlpatterns = [
    path('add/', views.add,name='add'),
    path('list/', views.list,name='list'),
    path('edit/(?P<id>\d+)/$', views.edit,name='edit'),
]
```

```
from django.shortcuts import render, redirect
from .models import *
from .forms import FeedbackForm

def add(request):
    if request.method=='POST':
        form = FeedbackForm(request.POST)
        if form.is_valid():
            form.save()
        return redirect('/feedback/list')
    else:
        form = FeedbackForm()

    return render(request, 'feedback.html', {'form': form})

def list(request):
    #return render(request, 'list.html')
    feedbacks = Feedback.objects.all()
    return render(request, 'list.html', {'feedbacks': feedbacks})

def edit(request,id):
    fb = Feedback.objects.get(pk=id)
    if request.method=='POST':
        form = FeedbackForm(request.POST, instance=fb)
        if form.is_valid():
            form.save()
        return redirect('/feedback/list')
    else:
        form = FeedbackForm(instance=fb)

    return render(request, 'feedback.html', {'form': form})
```

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css"
integrity="sha384-
MCw98/SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8
crossorigin="anonymous">
    <script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3
integrity="sha384-
ChfqqxuZUCnJSK3+MXmPNIyE6ZbWh2IMqE241rYiqJxyMiZ6OW/JmZQ
crossorigin="anonymous"></script>
</head>
<body>
     <div class="container">
        {% block title %}
        {% endblock title %}
    </div>

    <div class="container">
        {% block content %}
        {% endblock content %}
    </div>
</body>
</html>
```



```
{% extends "base.html" %}

{% block title %}
    <h3>Your Feedback..</h3>
{% endblock title %}

{% block content %}
    <p>
        <a href="{% url 'list' %}">Goto Feedback List</a>
    </p>

    <div>
        <form method="POST">
            {% csrf_token %}
            <table class="table">
            {{ form.as_table }}
            </table>
            <button type="submit">저장</button>
        </form>
   </div>

{% endblock content %}
```



```
{% extends "base.html" %}
{% load staticfiles %}

{% block title %}
    <h3>
        <img src="{% static 'images/note.jpg' %}" style="width:50px" />
        Feedback List
    </h3>
{% endblock title %}

{% block content %}
    <p>
        <a href="{% url 'add' %}">+ Create New</a>
    </p>

    <table class="table">
        <tr><th>Id</th><th>Name</th><th>Email</th><th>Comment</th></tr>
        {% for item in feedbacks %}
            <tr>
                <td>
                    <a href="{% url 'edit' id=item.id %}">{{ item.id }}</a>
                </td>
                <td>{{ item.name }}</td>
                <td>{{ item.email }}</td>
                <td>{{ item.comment }}</td>
            </tr>
        {% endfor %}
    </table>
{% endblock content %}
```